

Advanced Systems Lab

Understanding Throughput and Response Time

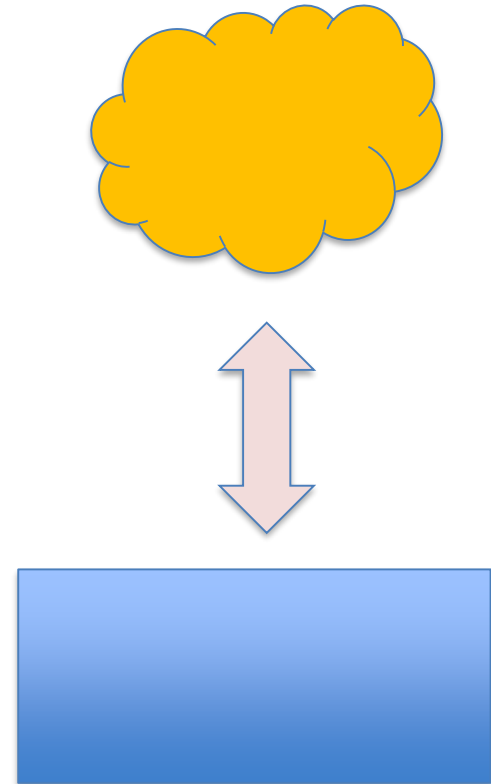
G. Alonso

Systems Group

<http://www.systems.ethz.ch>

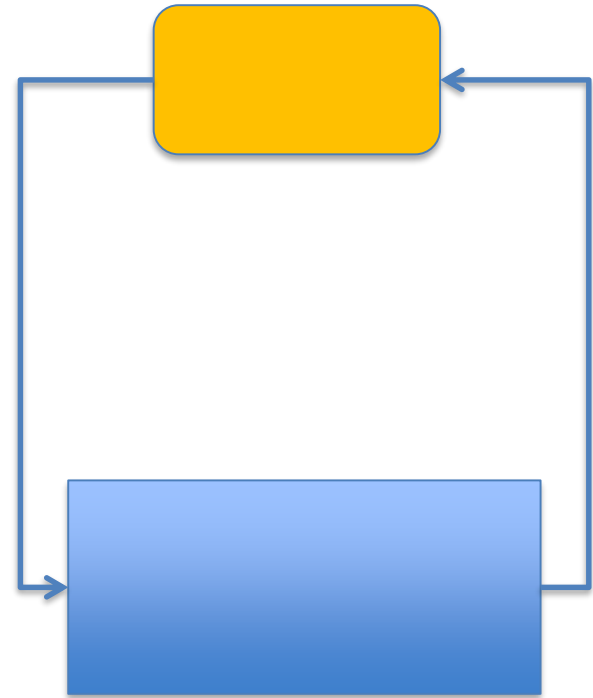
Open system

- Load comes from a potentially unlimited set of clients
- Load is not limited by clients waiting
- Load is not self-adjusting (load keeps coming even if system stops)
- Tests system's stability
- Example: web server



Closed system

- Load comes from a limited set of clients
- Clients wait for response before sending next request
- Load is self-adjusting
- System tends to stability
- Example: database with local clients



Example

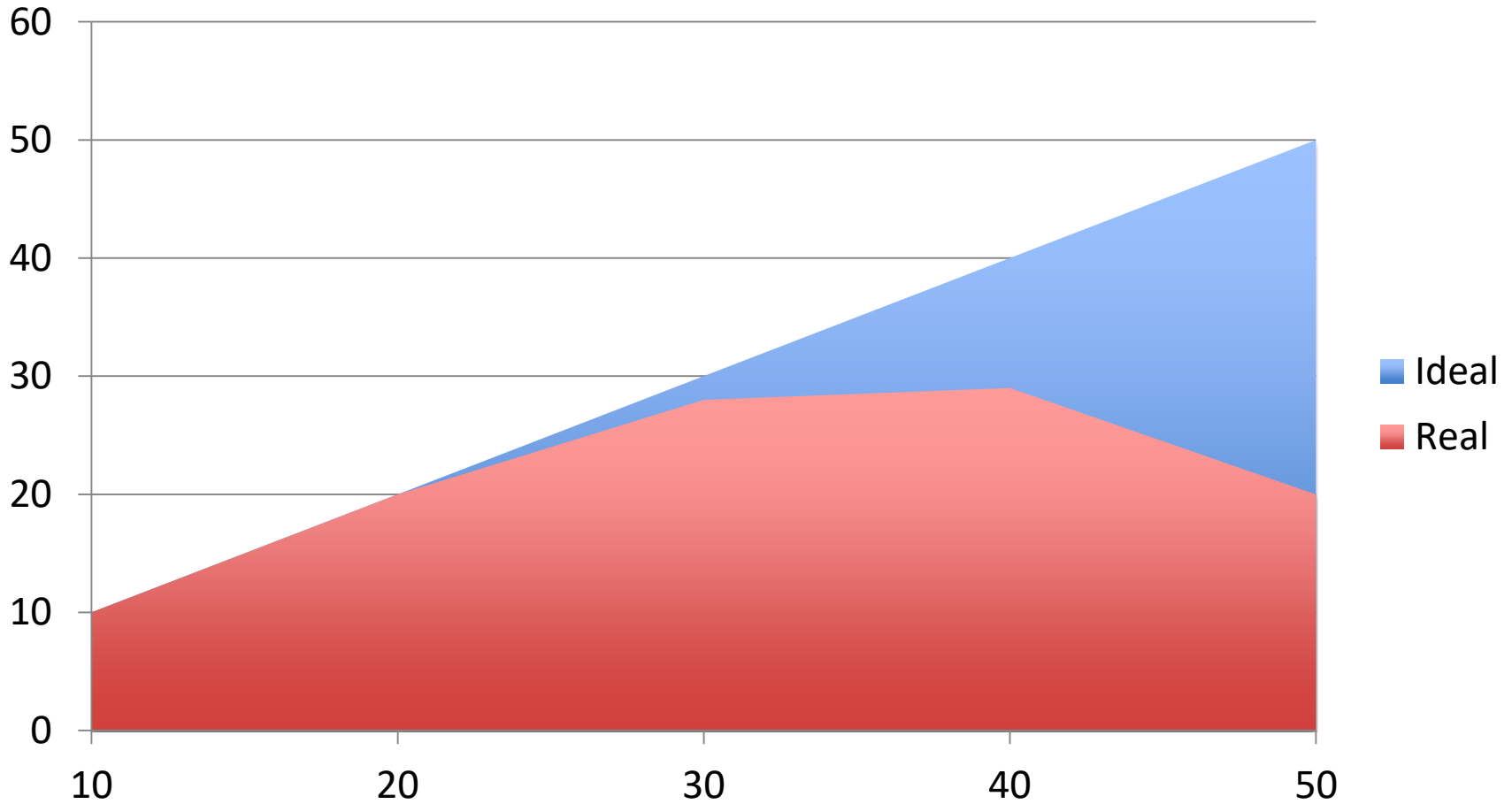
- Throughput: requests completed per unit of time (secs)
 - count only “successful” requests (no error, < timeout)
- Response Time: time to complete a given task when measured from a well defined start and end points
 - max/min/avg time (secs) to complete a request
 - time for “successful” and completed tasks

These are not absolute values, they depend on the load on the system:

- User Load: number of requests arriving per unit of time (secs)

How does throughput look like

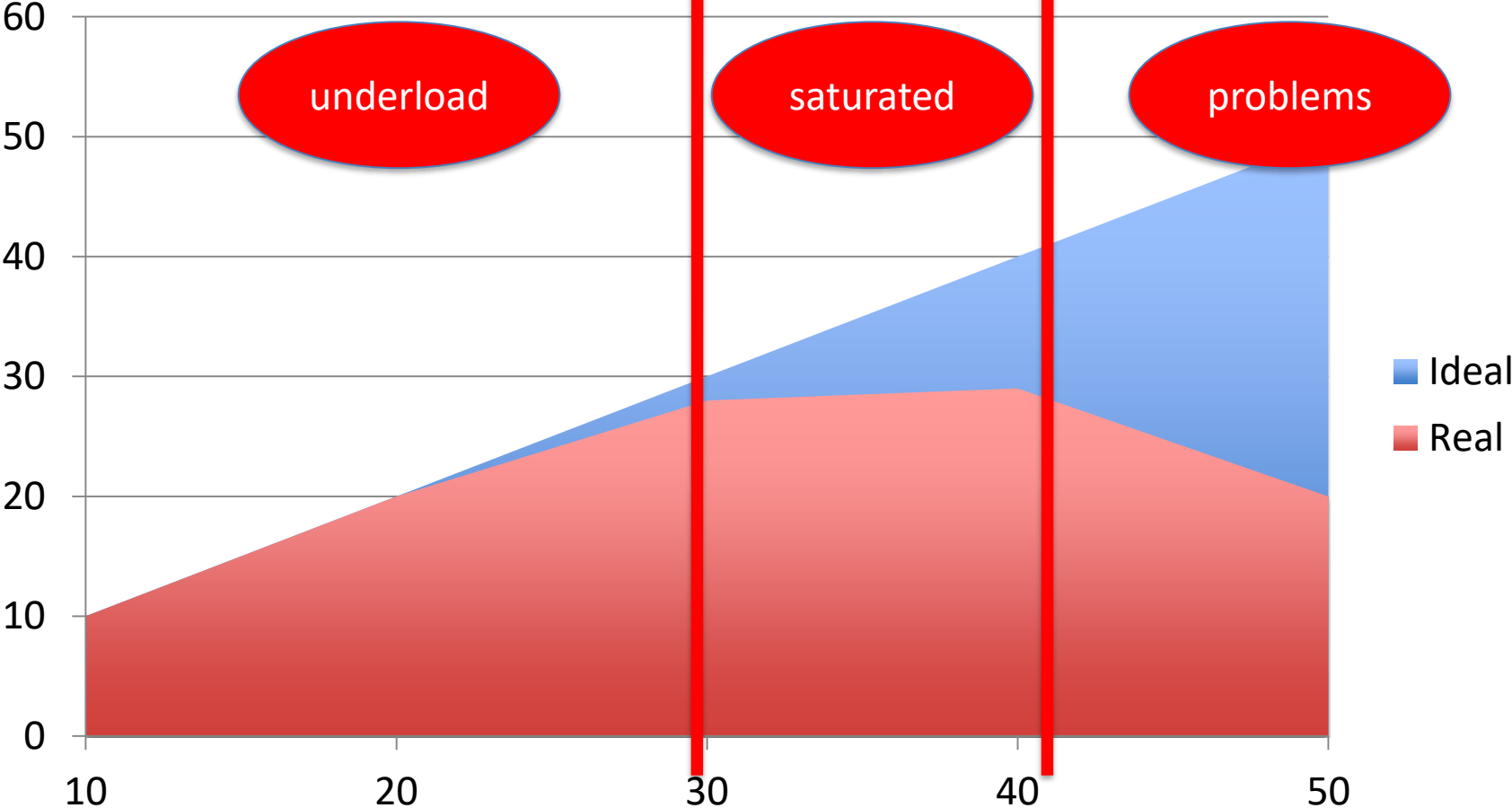
Throughput (req/sec)



User load (req/sec)

Throughput Analysis

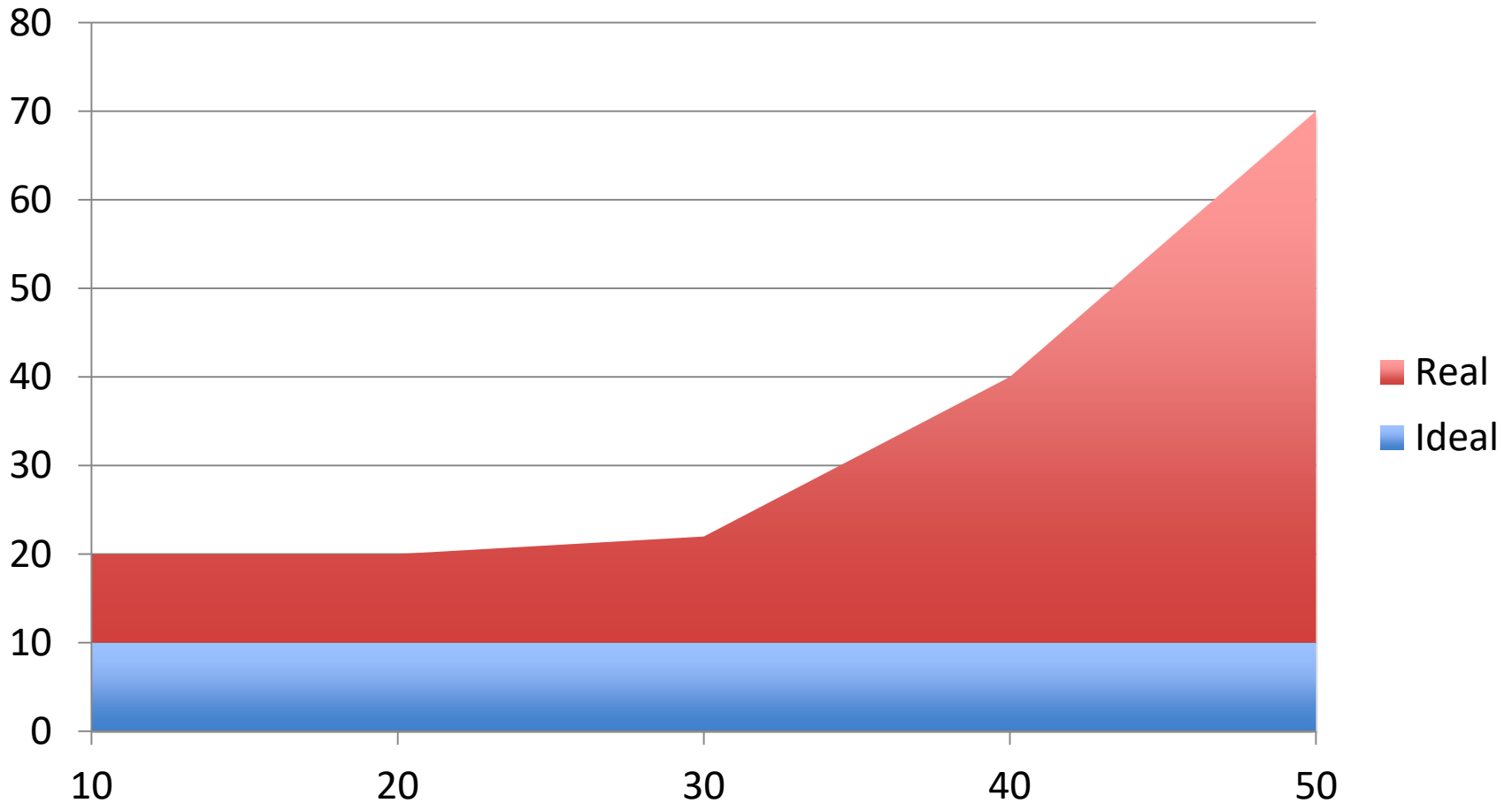
Throughput (req/sec)



User load (req/sec)

How does Response Time look like

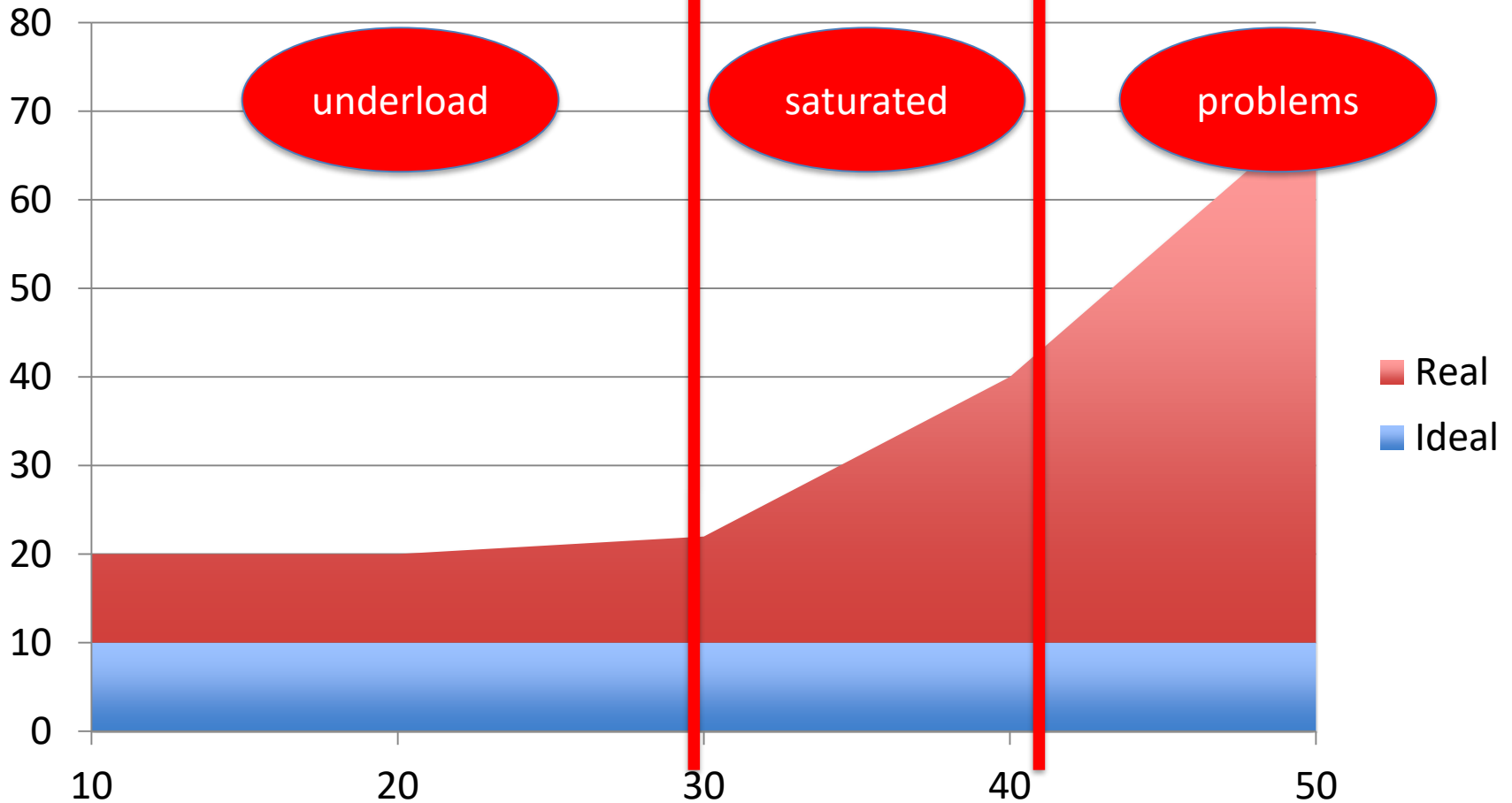
Response Time (sec)



User load (req/sec)

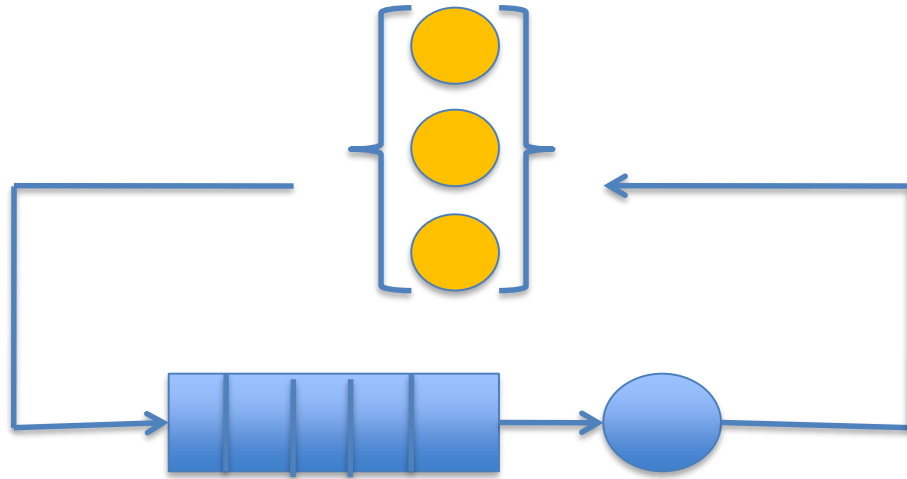
Response Time Analysis

Response Time (sec)



User load (req/sec)

Interactive Response Time Law



- Users submit a request, when they get a response, they think for a time Z , and submit the next request
- Applies to measurements in any system under a generic set of conditions

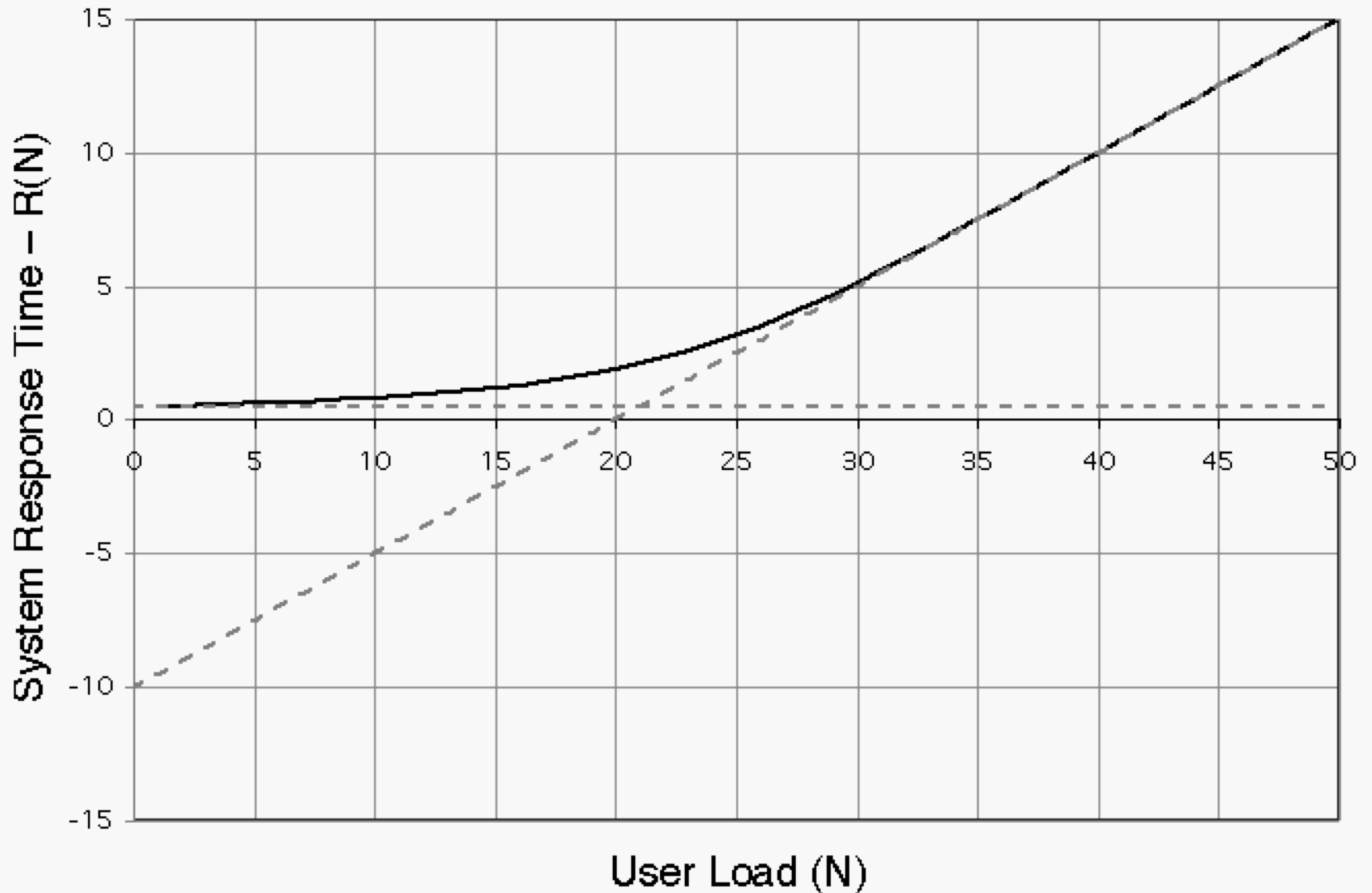
Interactive, closed systems

- Response time R
- Total cycle time $R+Z$
- Each user generates $T/(R+Z)$ requests in time T
- There are N users

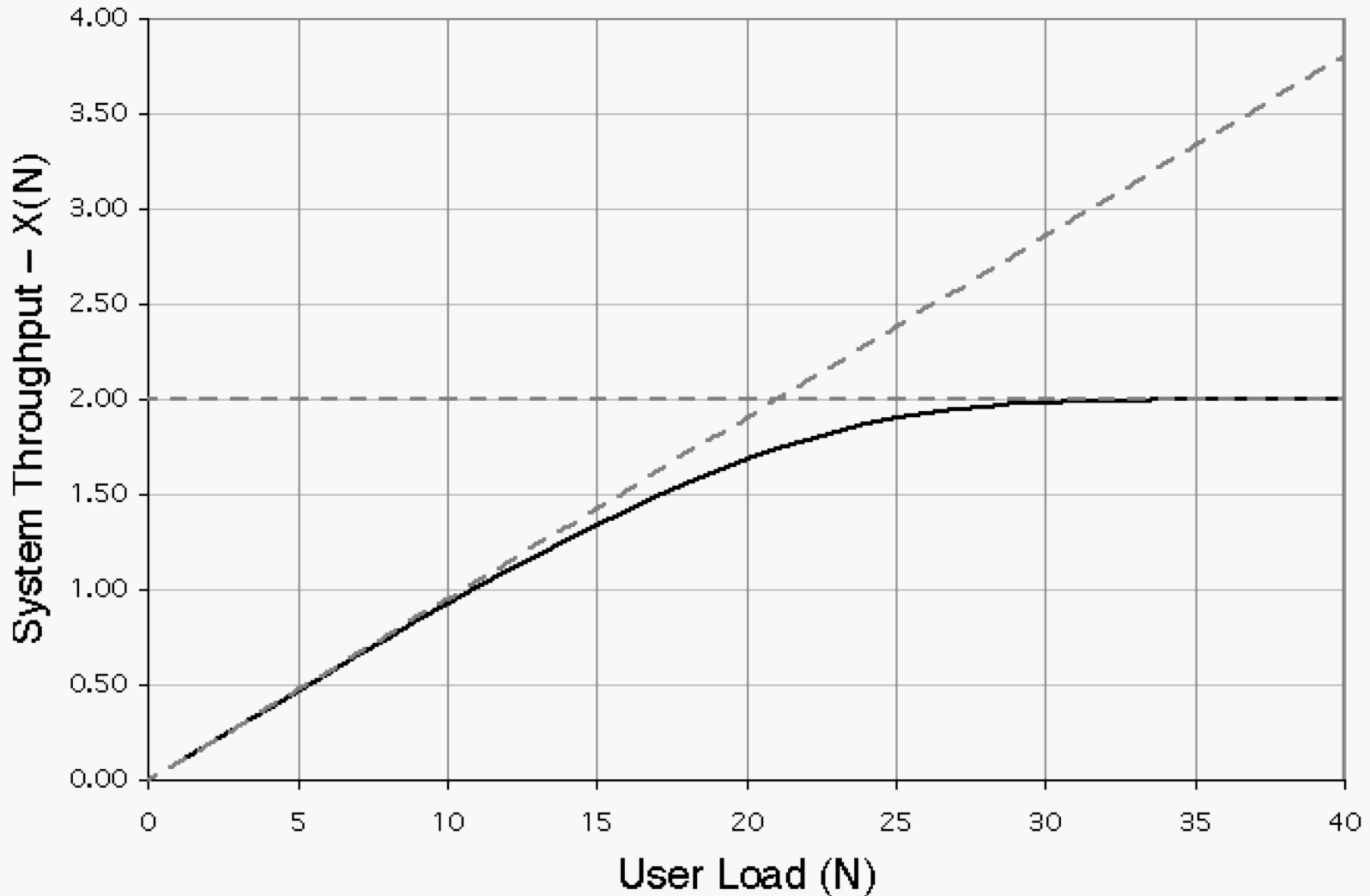
$$X = \frac{\text{Jobs}}{\text{Time}} = \frac{N \frac{T}{R+Z}}{T} = \frac{N}{R+Z}$$

$$R = \frac{N}{X} - Z$$

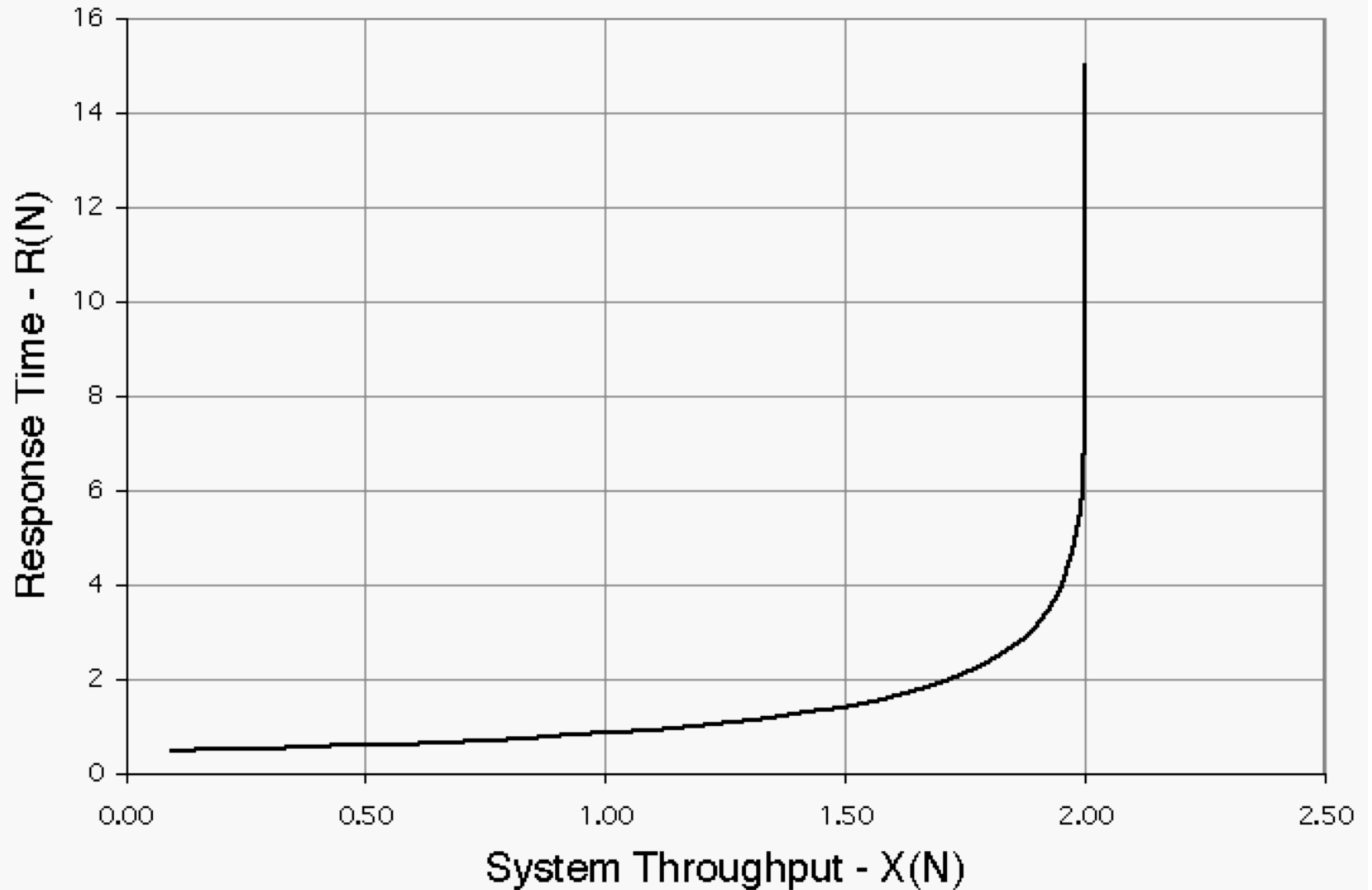
Interactive Law in practice



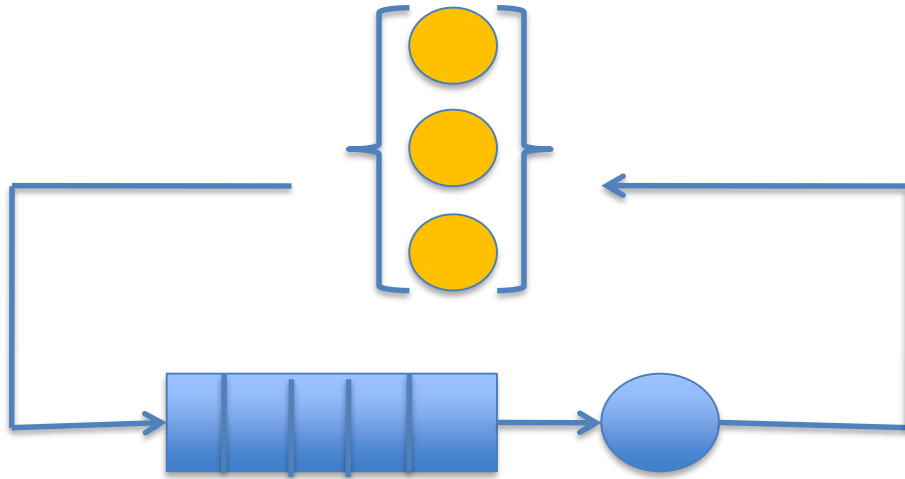
Interactive law in practice



Interactive law in practice

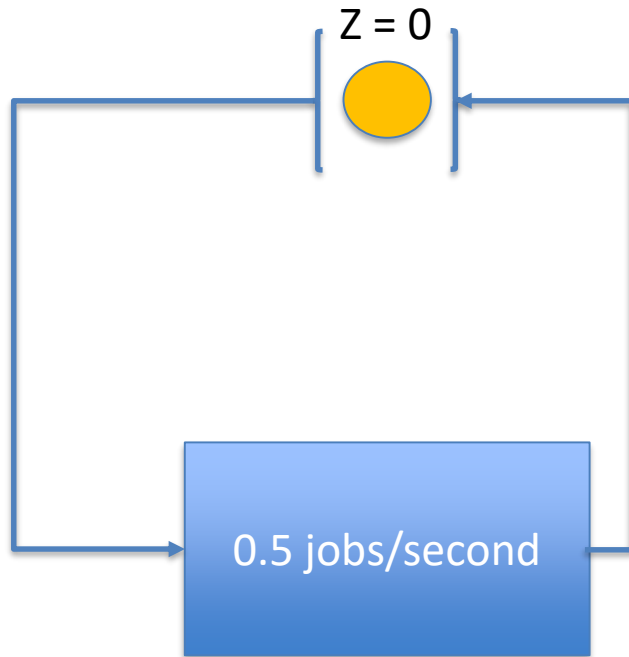


Warning!



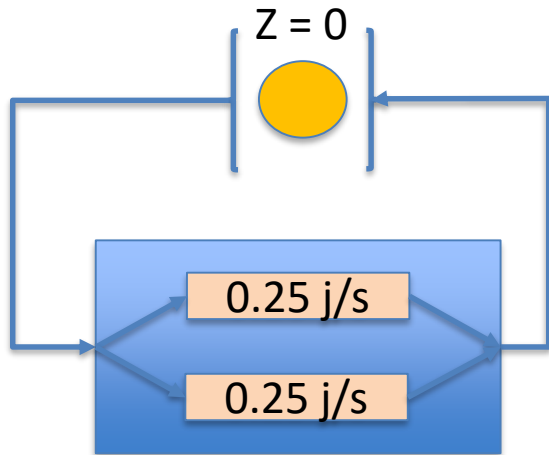
- Real systems are not ideal
 - Not all jobs are equal
 - Not all processing is identical
 - Network effects
 - Processing overhead (not only wait)
 - Exceptions and not “normal” return values

Example

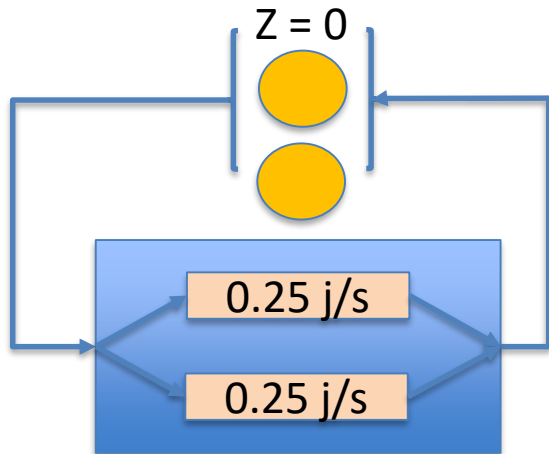


- What is the throughput?
- What is the response time

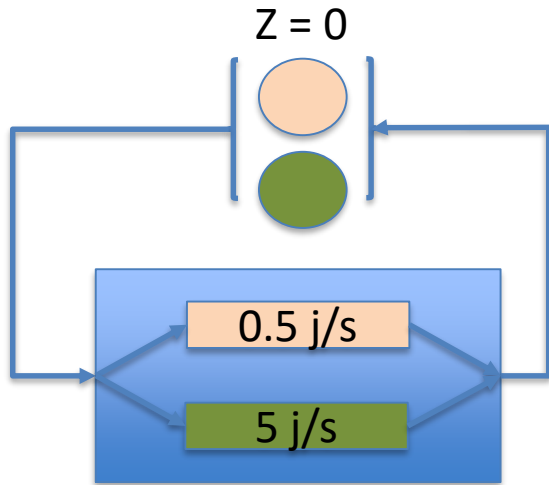
Example



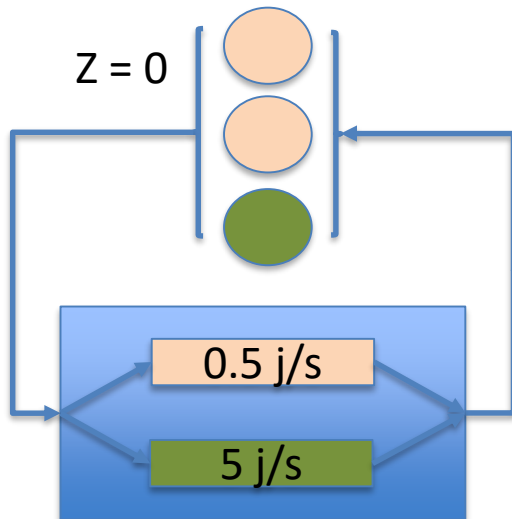
- What is the throughput?
- What is the response time
- Try for 1 clients, 2, 3, 4 ...
- Plot the results



Example

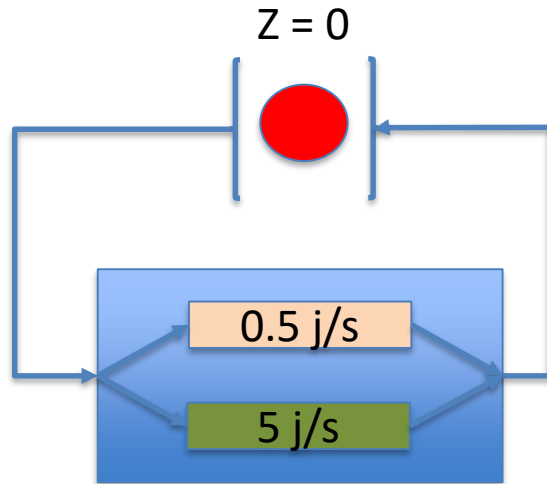


- Throughput is 5.5 j/s
- Response time is $(2/5.5)$ 0.36 seconds
- Actual response times are 2 sec for pink jobs and 0.2 sec for green jobs



- Throughput is 5.5 j/s
- Response time is $(3/5.5)$ 0.54 seconds
- Actual response times are 4 sec for pink jobs and 0.2 sec for green jobs

One more example



- Client submits 1 pink job and 1 green job, alternating:
 - Throughput = 2 jobs/2.2 sec = 0.9 j/s
 - RT => 1.1 seconds
- Client submits 1 pink job and 2 green jobs, and starts over again
 - Throughput = 3 jobs/2.4 sec = 1.25 j/s
 - RT => 0.8 seconds
- Client submits 2 pink jobs and 1 green job, and starts over again
 - Throughput = 3 jobs/4.2 sec = 0.71 j/s
 - RT => 1.4 seconds

Take home message

- What you measure (throughput and/or response time) depends on the workload
- How you submit jobs (and in which order) can make a big difference
- Understanding the performance of your system requires to understand how it works (the inner workings of the system) and the workload you are considering.